Logique de ressources et réseaux syntaxiques

Alain Lecomte Christian Retoré

INRIA-Lorraine & U. Grenoble 2
Bâtiment SHM
1251 Avenue Centrale
BP 47
38040 Grenoble cedex 9
Tél : (33)(0)4 82 78 52
Fax : (33)(0)4 82 56 65
email: lecomte@upmf-grenoble.fr

INRIA-Lorraine & CRIN-CNRS
Bâtiment LORIA
615 rue du Jardin Botanique
BP 101
54602 Villers les Nancy cedex
Tél : (33) (0)3 83 59 20 17
Fax : (33) (0)3 83 27 83 19
email:retore@loria.fr
http://www.loria.fr/~retore/index.html

Résumé

Le but de cet article est de développer un cadre logique commun pour exprimer des formalismes grammaticaux éloignés en apparence, comme les grammaires catégorielles (calcul de Lambek), les grammaires d'arbres adjoints, les grammaires basées sur gouvernement et liage et les grammaires "minimalistes" (au sens de Chomsky et de Stabler). Ce cadre est fourni par le calcul ordonné, extension non-commutative de la logique linéaire. Un outil fondamental dans cette approche est fourni par la notion de réseau de preuve. Au lieu d'être associées à des formules (comme dans les variantes du calcul de Lambek, et selon la théorie des types usuelle), les expressions sont associées à des preuves partielles (appelées aussi modules) et le calcul donne les manières de les composer. On fabrique, de manière générale, des grammaires de réseaux, ces réseaux ayant, du fait de la sémantique de la logique linéaire (même enrichie au connecteur "précède"), une interprétation en termes d'algèbre de processus.

1. Introduction

Deux notions en rapport avec la logique jouent aujourd'hui un rôle central dans les théories linguistiques : celle de *ressource* et celle de *dérivation*. Dans cet article, nous allons revenir sur l'importance de ces notions et montrer une manière de les traiter dans un cadre logique dérivé de la logique linéaire (J.Y. Girard, 1987, 1995).

2. Motivation pour un cadre dérivationnel et sensible aux ressources

Curieuse évolution que celle de la théorie syntaxique aujourd'hui : partie d'une conception générative accordant une place importante à la notion de règle syntagmatique (en opposition par exemple aux règles de réduction des grammaires catégorielles), elle a glissé peu à peu vers une théorie des "arbres admissibles" que l'on trouve aussi bien dans GPSG (Gazdar et al. 1985) que dans le module X-barre de la théorie chomskyenne, pour aboutir à une conception radicalement différente, présente chez N. Chomsky dernière manière (Chomsky, 1996) mais aussi chez des auteurs comme M. Steedman (Steedman, 1996), M. Johnson (Johnson, 1997) ou E. Stabler (Stabler, 1996), conception selon laquelle il n'est pas d'autre représentation de la structure d'une phrase que la pure et simple *preuve* de sa correction. De nombreux auteurs (dont Berwick & Epstein, 1996, Stabler, 1996) ont alors noté la convergence dans l'évolution entre la théorie syntaxique chomskyenne et les grammaires catégorielles. De fait, l'une des deux opérations que Chomsky nomme Transformations Généralisées, à savoir : *Merge*, n'est rien d'autre que l'application fonctionnelle propre aux grammaires catégorielles les plus simples (celles dites de "Ajduckiewicz-Bar-Hillel"). Toutefois, Chomsky introduit une autre TG : *Move*, dont le rôle est de déplacer un constituant lorsque se présente la nécessité de vérifier un type de trait. Par exemple, dans les langues SVO, à partir de l'hypothèse du sujet interne au SV, le sujet "monte" ("visiblement") dans la structure afin de se mettre en position de

vérification de ses traits d'accord et de cas. De même, le syntagme interrogatif, dans les langues non-*in situ*, est "attiré" par un trait fort porté par le noeud COMP. Dans toutes ces situations, il est devenu fréquent de dire que les traits en question sont *des ressources* et que la dérivation est terminée lorsque toutes les ressources ont été consommées (ou éventuellement "vérifiées"). C'est donc l'objectif de consommation des ressources qui guide la dérivation.

D'autres cadres théoriques utilisent désormais cette notion. On citera en particulier LFG. Dans les Grammaires Lexicales Fonctionnelles, on construit, à partir de règles syntagmatiques, une c-structure (ou structure en constituants) appariée avec une f-structure (ou structure fonctionnelle). Des contraintes agissent pour rendre ces représentations acceptables ou non. Ainsi le *principe de cohérence* impose-t-il que toutes les sous-f-structures correspondent bien à un prédicat qui les contrôle, et le *principe de complétude* que toutes les fonctions contrôlées par le prédicat soient bien présentes dans la f-structure. Dalrymple et al. (1995) enrichissent ce système de manière à pouvoir produire un autre niveau de représentation, ou s-structure, qui correspond à la représentation sémantique (ou pourrait-on dire par analogie avec la théorie chomskyenne, représentation "en forme logique"). Ils définissent pour celà une projection de la f-structure qui est une application traitant chaque composant de celle-ci comme une ressource utilisée une et une seule fois. Pour définir cette projection, on a besoin de transformer la f-structure en une formule de logique linéaire, combinant au moyen des connecteurs linéaires (⊗, --o), des formules de logique des prédicats d'ordre supérieur. Une logique de ressources est alors ici utilisée comme "langage d'assemblage des significations". M. Johnson (1996) a étendu cet usage de la logique linéaire à l'architecture générale de LFG : il en obtient une simplification notable. Dans sa conception, les règles syntagmatiques engendrent une c-structure étiquetée et l'interprétation sémantique provient directement de la preuve de la bonne formation du système des étiquettes, qui est en même temps preuve de complétude et de cohérence.

Dans un cadre strictement catégoriel enfin, il faut noter que M. Steedman (Steedman, 1996) donne une version de la plupart des modules de la théorie chomskyenne du genre "GB" sans postuler de niveaux de représentation intermédiaires, en particulier donc, sans déplacements ni catégories vides.

Là nous semble être l'apport essentiel de l'approche dite "dérivationnelle", en tant qu'opposée à "représentationnelle" : là où cette dernière exige plusieurs niveaux de représentation, avec des aller-retours de l'un à l'autre pour vérifier la bonne formation d'un énoncé, la première tente de construire en un seul geste et la preuve de bonne-formation et la représentation sémantique. Elle réduit également les variétés d'objets (genre "catégories vides") auxquels l'analyse a recours.

Nous cherchons dans ce qui suit à poser les bases d'une formalisation de ces approches, qui puisse devenir ensuite le point de départ à une implémentation rigoureuse des théories syntaxiques. Nous partirons donc de l'exposé du système le plus simple que nous puissions concevoir pour la déduction sensible aux ressources : le système de la logique linéaire multiplicative et intuitionniste.

3. D'une logique sensible à la quantité à une logique également sensible à l'ordre

3.1. Logique linéaire intuitionniste restreinte à {⊗, --o}

Les règles de LLI$_{\{, --o\}}$ (système noté aussi LP - pour "Lambek - permutationnel") sont les suivantes : (exprimées sous forme d'un calcul des séquents bilatère)

$$\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, A \otimes B, \Delta \vdash C}\,[g\otimes] \qquad \frac{\Gamma \vdash A \qquad \Gamma' \vdash B}{\Gamma, \Gamma' \vdash A \otimes B}\,[d\otimes]$$

$$\frac{\Gamma \vdash A \qquad \Gamma', B, \Delta' \vdash C}{\Gamma', \Gamma, A \multimap B, \Delta' \vdash C}\,[g\multimap] \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \multimap B}\,[d\multimap]$$

auxquelles nous ajoutons les *règles d'identité* : axiome (A |- A) et règle de coupure (transitivité), ainsi que, pour seule règle structurelle, la règle d' échange (pas de règle de contraction, ni d'affaiblissement donc).

Il est bien connu (van Benthem, 1986) que si nous décorons ces règles avec des *lambda-termes typés*, nous pouvons associer à toute preuve un lambda-terme qui est en quelque sorte la matérialisation de l'objet-preuve. Cette association se fait selon l'isomorphisme de Curry-Howard (entre types et propositions, λ-termes et déductions, la règle [g--o] étant associée à l'opération d' application, la règle [d--o] à l' abstraction, les règles pour $\otimes$ au couplage). Ce système (mais sans [d--o]) est utilisé par Dalrymple, Pereira et al. (Dalrymple et al. 1993) dans leur définition de la projection sémantique d'une f-structure en LFG.

Il est facile de montrer (se reporter par exemple à Retoré, 1996b) qu'on peut le plonger dans un calcul classique unilatère MLL (logique linéaire multiplicative) basé sur les deux connecteurs multiplicatifs $\otimes$ et $\wp$, à condition d'introduire une négation $()^{\perp}$ telle que l'on ait:

$(A^{\perp})^{\perp} \equiv A$ $\quad\quad$ $(A \wp B)^{\perp} \equiv A^{\perp} \otimes B^{\perp}$ $\quad\quad$ $(A \otimes B)^{\perp} \equiv A^{\perp} \wp B^{\perp}$

(une formule $F^{\perp}$ désigne alors la formule obtenue en transportant les symboles de négation sur les atomes au moyen de ces identités). L'implication linéaire --o se définit alors par :

$$A \multimap B \equiv A^{\perp} \wp B$$

Les règles de ce calcul sont données par :

$$\frac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, A \wp B, \Delta}\,[\wp] \qquad\qquad \frac{\vdash \Gamma, A \qquad \vdash \Gamma', B}{\vdash \Gamma, \Gamma', A \otimes B}\,[\otimes]$$

avec toujours règle d'échange, axiome (exprimé par : $\vdash A, A^{\perp}$) et règle de coupure

On a un critère pour reconnaître dans MLL les formules et les séquents qui sont les traductions possibles de formules et de séquents de LLI. (Les formules sont "polarisées" de telle sorte qu'un atome négatif soit de polarité négative, un atome positif de polarité positive, aucune sous-formule ne consiste à prendre soit le tenseur ($\otimes$) de formules négatives, soit le *par* ($\wp$) de deux formules positives, les séquents de LLI sont alors ceux qui ont une et une seule formule positive). Grâce à cela, on peut appliquer directement la méthode des **réseaux de preuve** de la logique linéaire afin d'obtenir des représentations géométriques des preuves effectuables dans le système.

3.2. Réseaux intuitionnistes

Il existe plusieurs conceptions des réseaux intuitionnistes (Girard, 1987, Danos & Régnier, 1990, Roorda, 1991, Lecomte, 1992, Lamarche, 1994). Nous retenons ici celle de C. Retoré (Retoré, 1996a, Retoré, 1996b) car elle se généralise ensuite facilement au calcul ordonné.

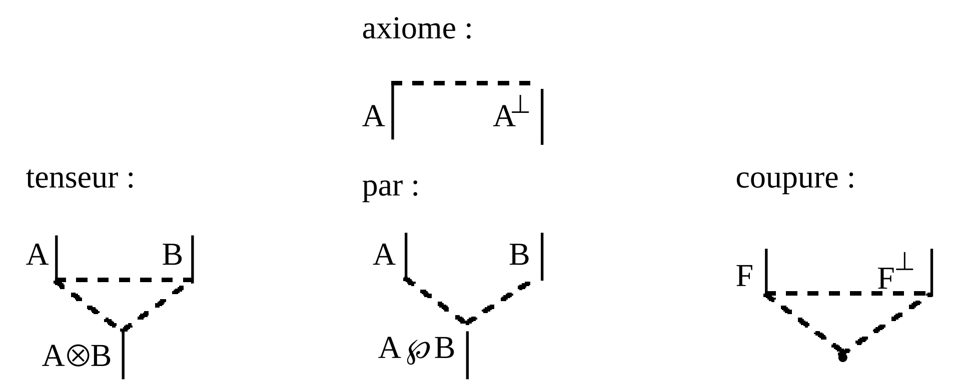
Définition 1 : un couplage est un sous-ensemble de l'ensemble des arêtes d'un graphe, tel que deux arêtes du couplage ne sont jamais adjacentes. Un tel couplage est dit parfait lorsque tout sommet est incident à une arête du couplage.

Définition 2 : un graphe R&B est un graphe dont les arêtes sont soit de couleur B (traits pleins), soit de couleur R (traits pointillés), tels que les arêtes B constituent un couplage parfait du graphe.

Les arêtes B correspondent aux formules et les arêtes R aux connecteurs. Dans ces graphes, l'identification de ceux qui correspondent à des démonstrations se fera au moyen de la notion de chemin alternant élémentaire.

Définition 3 : un chemin alternant élémentaire (Æ) dans un graphe R&B est un chemin dont les arêtes sont alternativement de couleur B et R et qui n'emprunte pas deux fois la même arête.

Définition 4 : les préréseaux sont des R&B graphes obtenus à partir de R&B graphes élémentaires appelés liens:

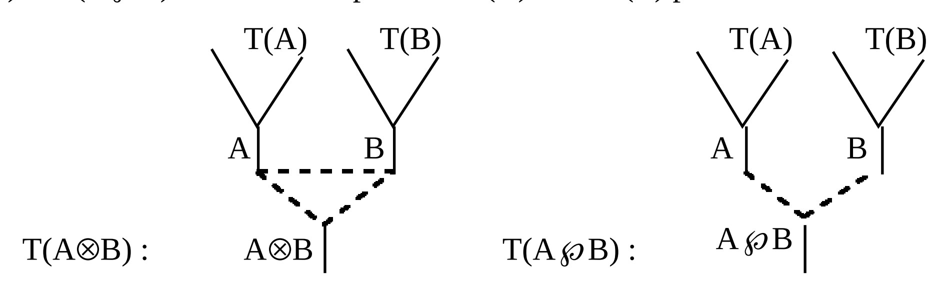


NB: dans ce dernier lien (coupure), F et $F^{\perp}$ sont des conclusions, le "•" est une marque indiquant l'occurrence de la règle de coupure (elle est en fait équivalente à une formule du second ordre $\exists X \; X \otimes X^{\perp}$).

Définition 5 : étant donnée une formule C, son arbre R&B des sous-formules T(C) est un graphe R&B défini inductivement comme suit :

• si C = α est atomique, alors T(C) est : | α

• $T(A \otimes B)$ et $T(A \wp B)$ sont définis à partir de T(A) et de T(B) par :



Définition 6 : un préréseau de conclusions Γ est constitué par la famille des arbres R&B des formules de Γ et par un ensemble d'arêtes R disjointes reliant des feuilles deux à deux orthogonales, de sorte que toute feuille soit atteinte exactement une fois.

Définition 7 : un réseau est un préréseau satisfaisant:

• non-Æ : absence de cycle alternant élémentaire

• SAT : entre deux points quelconques, il existe toujours un chemin Æ.

On peut alors démontrer:

Théorème 1 : un séquent S de MLL est prouvable si et seulement s'il existe un réseau dont les conclusions sont les formules de S.

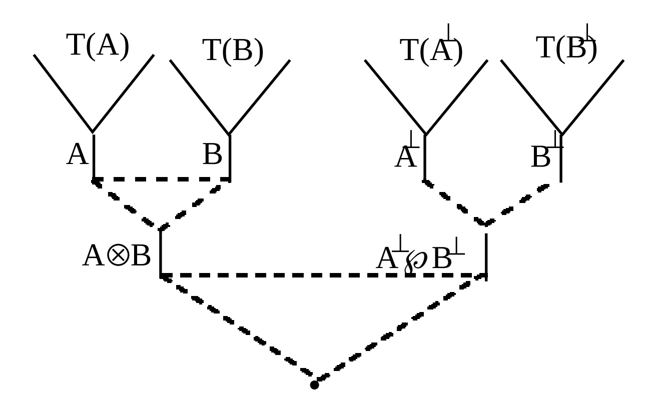
3.3. Élimination du lien coupure

Dans tout réseau possédant des liens coupure, il est possible d'éliminer ces derniers tout en gardant un réseau ayant les mêmes conclusions, à partir de deux pas élémentaires : si F et $F^{\perp}$ sont deux formules reliées par coupure, alors on peut avoir :

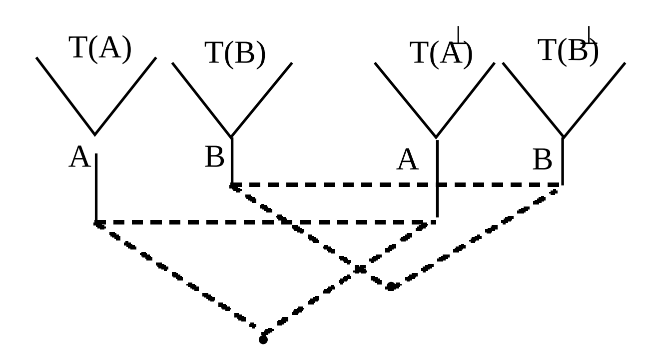
• F est un atome α, et donc $F^{\perp}$ l'atome négatif $\alpha^{\perp}$,

• F est une formule en $\otimes$: $A \otimes B$ et $F^{\perp}$ est sa formule duale $A^{\perp} \wp B^{\perp}$.

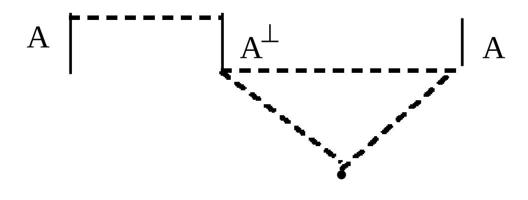
Le deuxième pas se représente par :



se réduit à :



et le premier :



se réduit à :



Définition 8 : un réseau est dit intuitionniste si et seulement si chaque conclusion reçoit une polarité et une seule conclusion est positive.

3.4. Réseaux de Lambek

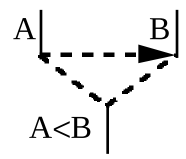
On obtient le calcul de Lambek (avec produit et séquence vide) à partir de LLI en supprimant la règle d'échange : cela a pour conséquence l'éclatement de --o en deux instances, qui sont orientées, et qu'on peut noter, selon la tradition des grammaires catégorielles : / et \, ainsi que la non-commutativité de ⊗. Les réseaux correspondants sont les réseaux intuitionnistes respectant en plus certaines conditions de planarité (Roorda, 1991, Lecomte, 1992, Retoré, 1996b). Nous ne nous étendrons pas davantage sur ces réseaux, désormais bien connus. Indiquons seulement qu'il s'agit de réseaux rendant compte d'une logique sensible non seulement à la quantité de ressources disponibles mais aussi à l'ordre dans lequel elles sont données. Toutefois, cette logique traite de structures *totalement* ordonnées, ce qui est excessif relativement aux applications linguistiques envisagées.

3.5. Réseaux ordonnés

On peut reprendre le calcul (classique) MLL et ajouter un nouveau type de lien.

Définition 9 : les préréseaux ordonnés sont des R&B graphes obtenus à partir des liens de la Définition 4, du lien suivant (le connecteur *précède* est noté "<"):

précède :



ainsi que d'arcs orientés (de type R) entre conclusions.

Remarque : dans ce qui suit, nous ne ferons pas usage des arcs orientés entre conclusions.

Définition 10 : les réseaux ordonnés sont les préréseaux satisfaisant:

• non-Æ : absence de *circuit* alternant élémentaire

• SAT : entre deux points quelconques, il existe toujours un chemin Æ.

Le calcul ainsi obtenu possède un certain nombre de propriétés remarquables, dont la préservation de l'invariant non-Æ par suppression du lien coupure.

3.6. Élimination du lien coupure dans les réseaux ordonnés

Le résultat est le même qu'au §3.3, en rajoutant un troisième pas pour le cas :

• F est une formule en "précède" : $A<B$ et $F^{\perp}$ est sa formule duale $A^{\perp}<B^{\perp}$

Quelques complications (non abordées ici, cf. Retoré, 1997) apparaissent pour les réseaux dotés, en plus, d'un ordre partiel sur les conclusions (arcs R entre conclusions et coupures).

3.7. Intérêt des réseaux de preuves

L'intérêt des réseaux de preuve est multiple :

- ils permettent de résoudre le problème souvent désigné comme celui des "ambiguïtés superflues" dans les grammaires catégorielles, c'est-à-dire l'existence de nombreuses déductions formellement différentes mais non sémantiquement différentes, et de ce fait ouvrent des perspectives nouvelles à la théorie de l'analyse syntaxique (Lecomte, 1993),

- ils fournissent un critère géométrique simple pour tester la validité d'une preuve,

- ils donnent une généralisation de la notion de λ-terme : il est en effet toujours possible de calculer le λ-terme de la preuve directement à partir du réseau dans LLI, et on peut définir les réseaux dans des contextes où la notion de λ-terme n'est pas définissable (en logique classique par exemple et dans le présent calcul ordonné), de ce fait, ils permettent d'étendre l'isomorphisme de Curry-Howard (cf. de Groote & Retoré, 1996).

3.7. Étiquetage des réseaux ordonnés

Dans le but de projeter l'ordre induit par le connecteur "précède" (qui correspond à un ordre d'évaluation des formules) sur l'ordre des mots, nous procédons à un étiquetage des réseaux : les formules associées aux arêtes B sont étiquetées par des multi-ensembles de chaînes partiellement ordonnés, selon les conventions suivantes :

• <u>lien axiome</u> : deux conclusions $(A, \mathfrak{R})$ et $(A^{\perp},\mathfrak{R}')$ telles que $\mathfrak{R} = \mathfrak{R}'$.

• <u>lien tenseur</u> : si la conclusion est étiquetée par $\mathfrak{R}$, l'une des deux prémisses est étiquetée par l'union de $\mathfrak{R}$ et du multi-ensemble $\mathfrak{R}'$ étiquetant l'autre prémisse (donc même étiquette pour les deux prémisses si $\mathfrak{R} = \emptyset$).

• <u>lien par</u> : si les prémisses sont étiquetées par $\mathfrak{R}$ et $\mathfrak{R}'$, la conclusion est étiquetée par $\mathfrak{R}\cup\mathfrak{R}'$.

• lien précède : si les prémisses sont étiquetées par $\Re$ et $\Re'$, la conclusion est étiquetée par $\Re<\Re'$.

• lien coupure : comme le lien tenseur, avec "•" étiqueté Ø.

4. Calcul ordonné et TALN

4.1. Réseaux ordonnés étiquetés pour l'ordre des mots

L'application du calcul ordonné au traitement de la langue naturelle se justifie particulièrement par le fait que dans la plupart des langues, des propriétés sont exprimées à partir de phénomènes d'ordre des mots, ainsi par exemple de la distinction des fonctions grammaticales dans les langues configurationnelles. D'autre part, des contraintes d'ordre existent, plus ou moins lâches, qui conduisent à considérer qu'une phrase est, par elle-même, un "multi-ensemble partiellement ordonné". Par exemple, en français, dans une relative, sujet et verbe peuvent être permutés, mais ce n'est pas le cas dans l'absolu :

(1) a. *l'homme que Marie aime*

b. *l'homme qu'aime Marie*

(2) *a. Marie aime un homme*

b. **Aime Marie un homme*

L'application du calcul ordonné au traitement de la LN se fait donc de la manière suivante:

1- les propositions (voire dans un système plus riche intégrant des variables individuelles, les prédicats) sont les catégories syntaxiques usuelles (qu'on peut choisir selon ses préférences évidemment, c'est-à-dire : d, n pour déterminant et nom, dp ou np pour le groupe nominal etc.),

2- les sous-formules sont étiquetées par des ensembles partiellement ordonnés de formes phonologiques, qui ne font qu'enregistrer la structure de la preuve dans l'ordre des mots, de sorte qu'en sortie du réseau, on possède un et un seul noeud de sortie (noté $s^{\perp}$) qui reçoit comme étiquette l'ordre (partiel) des mots correct de la phrase analysée.

4.2. Branchement de modules

Désormais, nous considérons qu'à tout item lexical correspond un réseau de preuve partielle, c'est-à-dire un préréseau vérifiant le critère d'absence de circuit alterné mais avec certaines feuilles non reliées par un lien axiome à d'autres feuilles (que nous appelons donc "hypothèses"). De tels réseaux partiels sont aussi appelés *modules*.

Il existe deux manière de "brancher" ensemble deux modules :

• soit par des liens axiomes : des hypothèses appartenant aux deux modules différents sont reliées par lien-axiome,

• soit par coupure + élimination de la coupure : deux conclusions duales l'une de l'autre appartenant chacune à un module différent sont reliées par un lien coupure, puis ce lien est supprimé en appliquant l'algorithme d'élimination de la coupure.

4.3. Exemples

Exemple 1 : (*grammaires catégorielles de type AB ou grammaires minimalistes avec seulement opération Merge*)

(3) *Pierre lit un livre :*

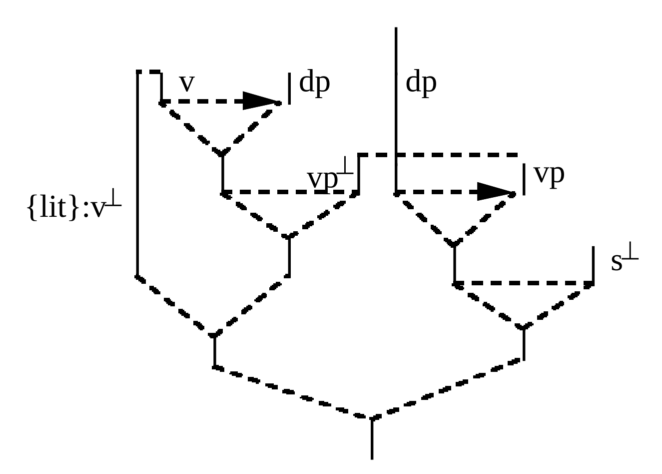
types : nous appelons *type d'une expression* une formule de logique linéaire ordonnée associée à cette expression, qui en donne la catégorie et le "mode d'emploi" + des liens axiomes

déjà introduits – c'est en ceci qu'il s'agit d'une preuve partielle – ici marqués par des indices entiers de sorte que deux atomes reliés par un lien axiome aient le même indice[1].

dp : {pierre}; (v[1] : {lit}) ⊗ ((v[1] < dp) --o vp[2])⊗((dp < vp[2]) --o s)

d: {un} ; (n[1] : {livre}) ⊗ ((d<n[1]) --o dp)

modules : le module est déduit du type par dualité : il faut en effet bien noter que le calcul ordonné est basé sur un calcul unilatère c'est-à-dire où toutes les formules sont en partie droite, ainsi un type A --o B devient par dualité : A⊗B$^{\perp}$, donnons par exemple le module entièrement déployé associé à l'expression "lit" :



qui pourra se noter "linéairement":

((v$^{\perp}$[1] : {lit}) ℘ ((v[1] < dp)⊗vp$^{\perp}$[2])) ℘ ((dp < vp[2])⊗s$^{\perp}$)

Dans cet exemple, pour des raisons de lisibilité, nous donnons chaque pas de dérivation du réseau final à partir des types, il faut alors bien penser que la vérification de la bonne formation (absence de cycle alternant) se fait toujours sur les modules obtenus par dualité.

branchements (ici, seulement par axiomes) :

pas 1 : (en gras : les nouveaux liens introduits à chaque étape, en italique : ce qui en résulte à chaque étape)

(d[**2**]: {un}) ; (n[1] : {livre}) ⊗ ((d[**2**] <n[1]) --o (dp : {*un < livre*})

pas 2 :

(v[3] : {lit}) ⊗ ((v[3] < dp[**5**]) --o (vp[4] : {*lit < un < livre*}) ⊗ ((dp < vp[4]) --o s)

; (d[2]: {un}) ; (n[1] : {livre}) ⊗ ((d[2] <n[1]) --o (dp[**5**] : {un < livre})

pas 3 :

(dp[**6**] : {pierre}) ;

(v[3] : {lit}) ⊗ ((v[3] < dp[5]) --o (vp[4] : {lit < un < livre}) ⊗ ((dp[**6**] < vp[4]) --o s : {*pierre<lit<un<livre*})

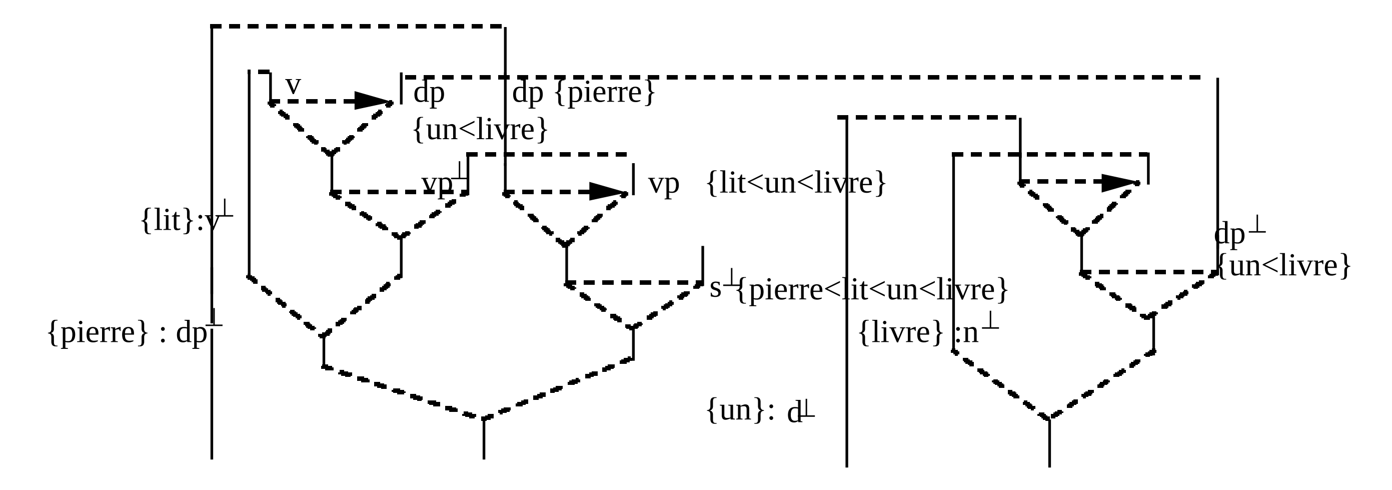
; d[2]: {un} (n[1] : {livre}) ⊗ ((d[2] <n[1]) --o (dp[5] : {un < livre})

Évidemment, la phrase en entrée est correctement analysée si et seulement si elle contient l'ordre déterminé par l'analyse (c'est-à-dire étiquetant la "sortie" s$^{\perp}$).

Le réseau final est le suivant :

---

[1] Ces indices n'ont évidemment pas de valeur par eux-mêmes, tout module obtenu à partir d'un module donné en changeant les valeurs des entiers lui resterait identique, pourvu que les mêmes atomes demeurent identifiés. Ceci explique que nous pourrons procéder à toute opération de renommage des indices.



Exemple 2 : (*Grammaires de type GB*)

Cet exemple traitera des clitiques, problème que l'on rencontre classiquement dans l'analyse des langues romanes. Chomsky (1981) fait l'hypothèse que dans une phrase comme (4) :

(4) *Pierre le voit*

on a une D/S-structure du genre : Pierre [$_{VP}$ le voit [$_{NP}$ e]], ceci afin que le θ-rôle d'objet de *voit* soit assigné, puis transmis à *le*. Nous ne faisons appel ici à aucune espèce de catégorie vide (d'où l'inutilité que nous nous prononcions en faveur d'une trace ou d'un PRO...), néanmoins nous proposons une analyse voisine, dans la mesure simplement où ce que Chomsky exprime par un lien problématique entre un élément et un vide, nous l'exprimons directement par un lien-axiome entre deux "ports" de nos modules qui se correspondent. Comme d'autre part, le clitique *le* sert non seulement à recevoir un θ-rôle objet du verbe, mais qu'il modifie le verbe lui-même en le transformant en une forme "cliticisée" (ici : *le-voit*), nous avons besoin de relier le module-clitique au module-verbe par deux opérations : l'une de branchement par coupure, l'autre de branchement par axiome. Le branchement par coupure n'est possible que si les deux modules ont des conclusions qui sont duales l'une de l'autre. Ceci nécessite de modifier les modules verbaux de manière à leur adjoindre une conclusion vide pouvant servir pour un tel branchement. Il y aura autant de telles conclusions que le verbe pourra admettre de clitiques. De plus, l'ordre dans lequel nous les installons au sein du module verbal n'est pas neutre : il permet de respecter les contraintes d'ordre sur les clitiques en français. Dernière remarque: l'établissement d'un lien coupure entre une conclusion vide du module clitique et la conclusion duale dans le module verbal ne pourra conduire à un réseau correct que si, parallèlement, le lien axiome entre l'hypothèse correspondant au cas du clitique et celle correspondant à l'assignation de cas du verbe peut être installé. Ceci élimine toute construction impossible du genre : **Pierre me lui lit*.

modules :

pierre : dp(Cas) : {pierre}

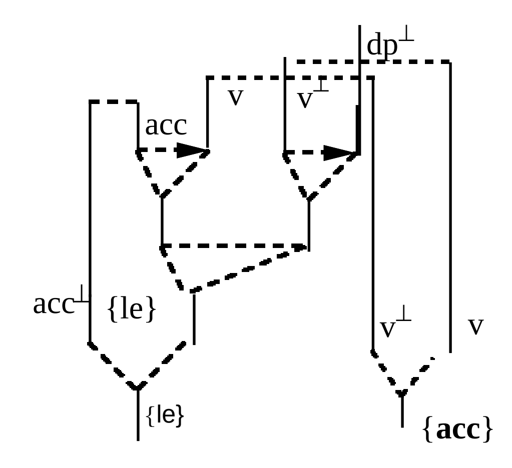
voit : $((v[1] \otimes v^{\perp}[2]) : \mathbf{acc})$;
$(v^{\perp}[1] : \{voit\}) \wp ((v[2] < dp(acc)) \otimes vp^{\perp}[3]) \wp ((dp(nom) < vp[3]) \otimes s^{\perp})$

le : $(acc^{\perp}[1] : \{le\}) \wp ((acc[1]<v[2]) \otimes (v^{\perp}[3]<(dp^{\perp}(acc) : \emptyset))$; $((v^{\perp}[2] \wp v[3]): \mathbf{acc})$

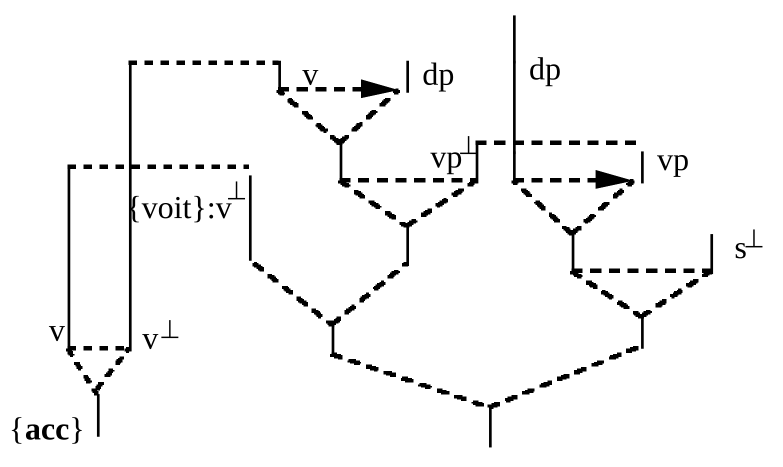
commentaire : le verbe fournit à la fois son entrée ordinaire et une conclusion supplémentaire : $(v \otimes v^{\perp})$, marquée **acc** pour accusatif. Le clitique contient la conclusion duale: $(v^{\perp} \wp v)$, avec la même marque, son autre conclusion exprimant (après conversion en formule duale!) le fait qu'il s'agit d'une marque accusative et que la séquence constituée de cette marque

suivie d'un verbe fournit l'équivalent d'un verbe suivi par un complément nominal accusatif (étiqueté "vide").

module "clitique-le" déployé :



module verbal avec une "entrée-clitique":



branchements :

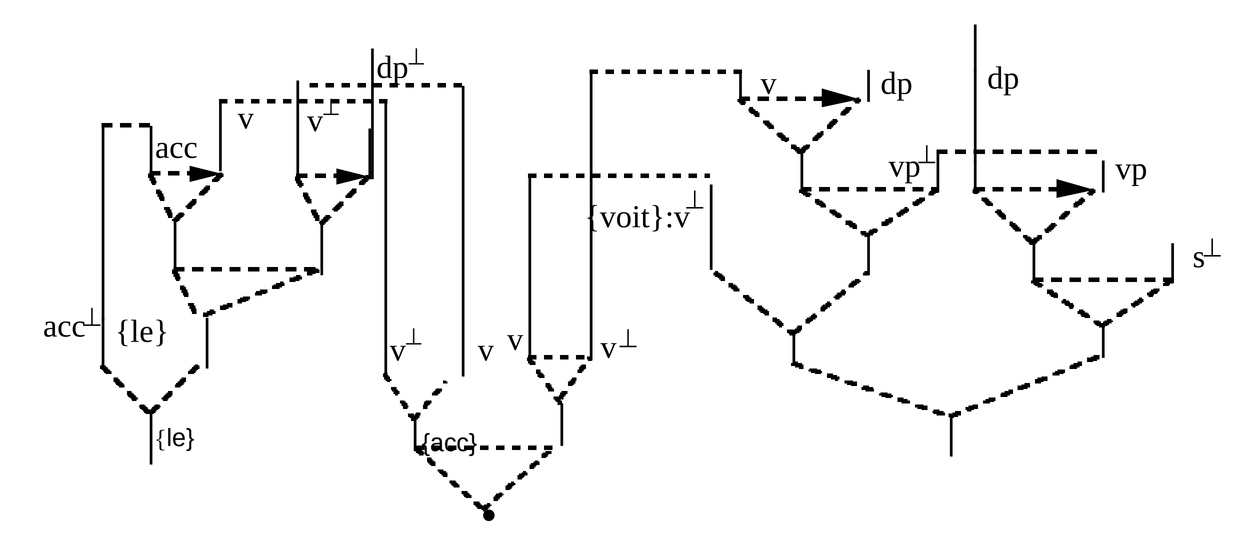
Pas 1 : (coupure)

(acc$^\perp$[1] : {le}) ℘ ((acc[1]<v[2]) ⊗ (v$^\perp$[3]<(dp$^\perp$(acc) : Ø)) ; ((v$^\perp$[**2**] ℘ v[**3**]): **acc**)

((v[**2**] ⊗ v$^\perp$[**3**]) : **acc**) ;

(v$^\perp$[2] : {voit}) ℘ ((v[3] < dp(acc)) ⊗ vp$^\perp$[4]) ℘ ((dp(nom) < vp[4]) ⊗ s$^\perp$)

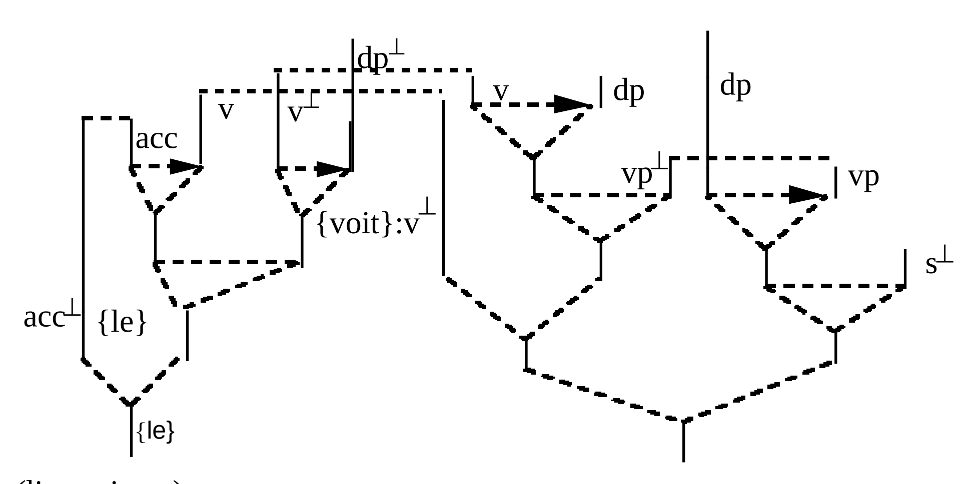
réseau :



<u>Pas 2</u> : (élimination de la coupure)

$(acc^{\perp}[1] : \{le\}) \wp ((acc[1]<v[\mathbf{2}]) \otimes ((v^{\perp}[\mathbf{3}] : \{\mathit{le<voit}\}) <(dp^{\perp}(acc) : \emptyset))$;

$(v^{\perp}[\mathbf{2}] : \{voit\}) \wp ((v[\mathbf{3}] < dp(acc)) \otimes vp^{\perp}[4]) \wp ((dp(nom) < vp[4]) \otimes s^{\perp})$

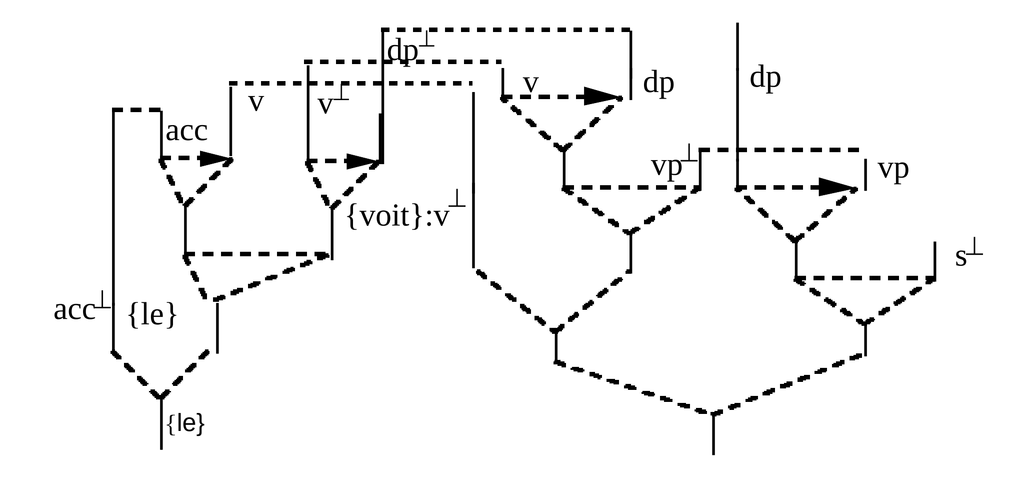


<u>Pas 3</u> : (lien axiome)

$(acc^{\perp}[1] : \{le\}) \wp ((acc[1]<v[2]) \otimes ((v^{\perp}[3] : \{le<voit\}) <(dp^{\perp}(acc)[\mathbf{5}] : \emptyset))$;

$(v^{\perp}[2] : \{voit\}) \wp (((v[3] : \{le<voit\}) < dp(acc)[\mathbf{5}]) \otimes (vp^{\perp}[4] : \{le<voit\})) \wp$

$((dp(nom) < vp[4]) \otimes s^{\perp})$



Pas 4 :

dp[**6**](nom) : {pierre} ;

$(acc^{\perp}[1] : \{le\}) \wp ((acc[1] < v[2]) \otimes ((v^{\perp}[3] : \{le < voit\}) < (dp^{\perp}(acc)[5] : \emptyset))$;

$(v^{\perp}[2] : \{voit\}) \wp (((v[3] : \{le < voit\}) < dp(acc)[5]) \otimes (vp^{\perp}[4] : \{le < voit\})) \wp$

$((dp(nom)[\mathbf{6}] < vp[4]) \otimes (s^{\perp} : \{\textit{pierre} < \textit{le} < \textit{voit}\}))$

Remarque 1 : au cas où il n'y a pas de clitique mais un complément réalisé par un NP lexicalement plein, v[1] et v[2] de l'entrée verbale sont identifiées [2], ce qui conduit alors à une analyse semblable à celle de l'exemple 1.

Remarque 2: afin d'éviter certaines constructions impossibles comme :

(5) **Pierre le croit que Marie lit* (au lieu de : *Pierre croit que Marie le lit*)

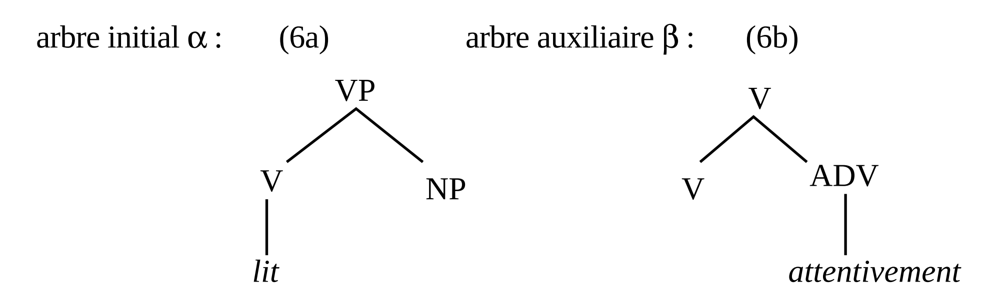
c'est-à-dire la cliticisation sur un verbe d'une phrase matrice, on doit rajouter aux réseaux des contraintes "non logiques", donc spécifiquement linguistiques. Ces contraintes ne sont rien d'autre que des critères de minimalité (condition sur les cycles clitique-clitique ne passant pas par un lien-axiome (s', $s'^{\perp}$)) qui sont dans la ligne de l'évolution actuelle des théories linguistiques (cf. en particulier Chomsky, 1996).

Exemple 3 : (*Grammaires de type LTAG*)

Une Grammaire d'Arbres Adjoints Lexicalisée est définie à partir de deux ensembles d'arbres dits *élémentaires* : les arbres *initiaux* et les arbres *auxiliaires*. Ces arbres ont tous au moins une feuille étiquetée par un noeud terminal (c'est pour cette raison qu'on les dit lexicalisés). Un arbre auxiliaire possède une feuille possédant la même étiquette que la racine (et sert ainsi à exprimer l'opération d'adjonction). Le langage engendré par une grammaire TAG *G* est la fermeture de l'ensemble des arbres élémentaires sous deux opérations : *substitution* et *adjonction*. La substitution est bien connue et facile à définir : on remplace un noeud étiqueté par un non-terminal et qui est sans descendant par un arbre dont la racine est étiquetée par le même non-terminal. L'adjonction consiste à "couper" un arbre α où a lieu l'adjonction en un noeud N et à insérer entre les deux parties issues de cette découpe un arbre auxiliaire β dont la racine est étiquetée N.

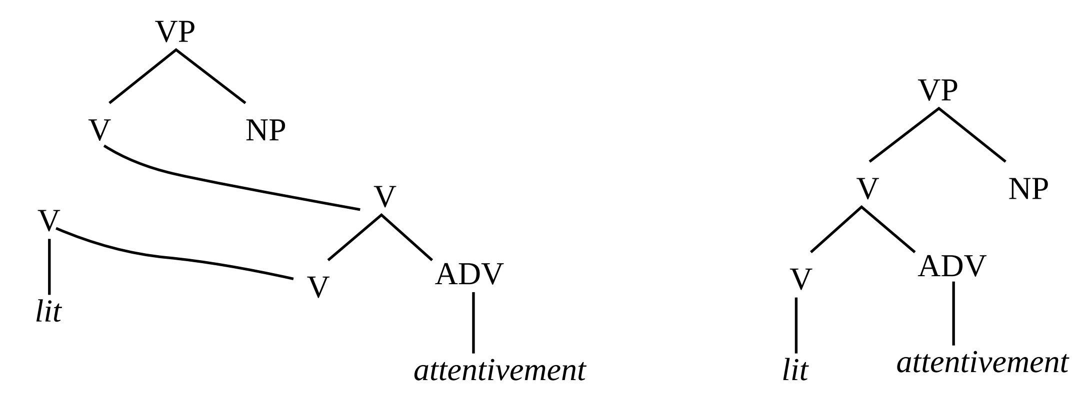
Exemple :

arbre initial α : (6a) arbre auxiliaire β : (6b)



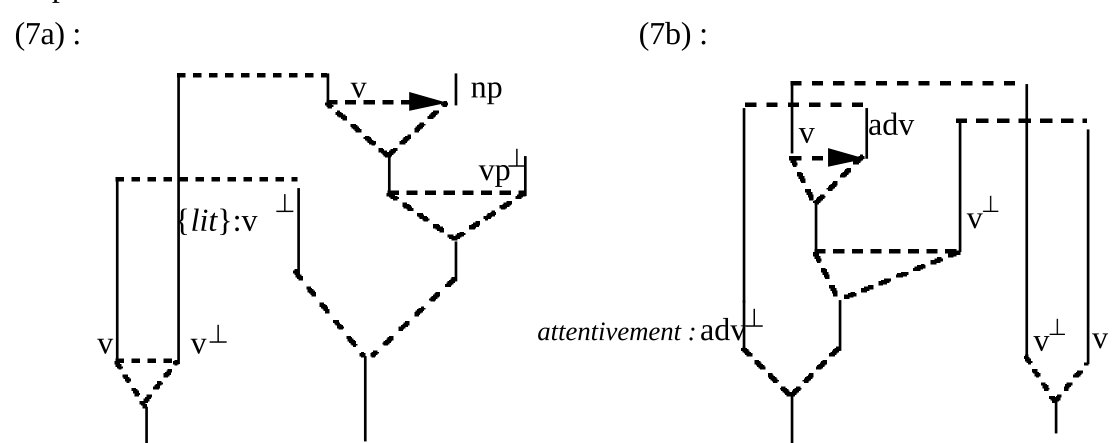
[2] Rappelons (§3-2) que "•' est équivalent à : $\exists X\ X^{\perp} \otimes X$, ce qui fait qu'une formule a $\otimes a^{\perp}$ peut toujours être considérée comme une coupure entre a et $a^{\perp}$, et l'élimination de la coupure dans ce cas précis revient à faire disparaître une telle conclusion.

L'adjonction des deux arbres se fait selon : qui donne : (6c)

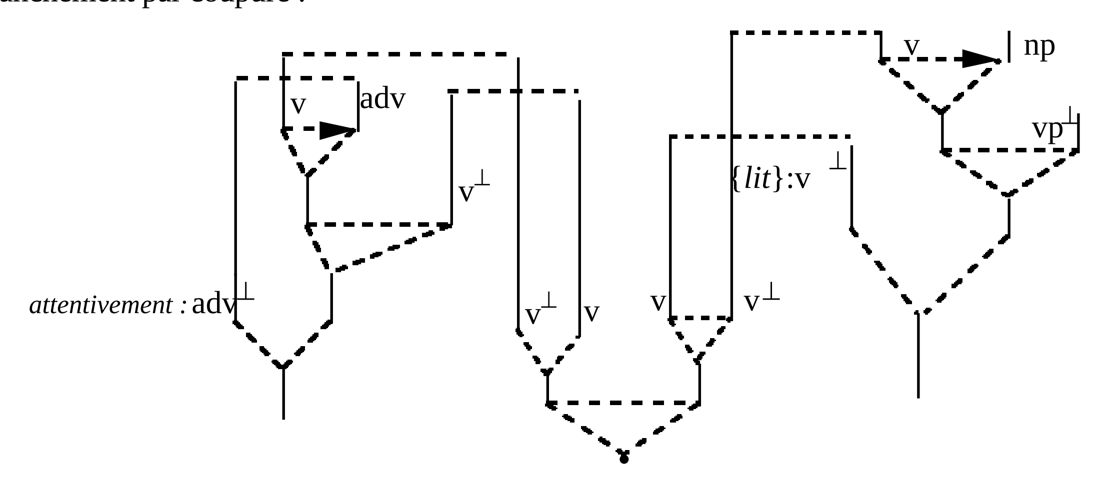


On peut montrer qu'en adoptant la traduction suivante (traduction des arbres élémentaires d'une grammaire *G* en réseaux de preuve partiels), on obtient dans le calcul ordonné, un ensemble de réseaux fermé sous branchement-axiome et branchement-coupure qui correspond à l'ensemble des arbres engendrés par *G*. Les noeuds des arbres sont représentés par des liens axiomes (le noeud vp par exemple par le lien axiome vp - $vp^{\perp}$). Un noeud x où une adjonction peut avoir lieu est représenté par l'introduction d'une conclusion $x \otimes x^{\perp}$ telle que le lien axiome initial $x\text{-}x^{\perp}$ représentant x soit scindé en deux liens axiomes, chacun reliant l'un des deux atomes de ce lien initial à l'atome correspondant de la conclusion introduite. Les traductions des arbres auxiliaires possèdent dualement une conclusion $x^{\perp} \wp x$. On obtient pour le même exemple :

(7a) : (7b) :

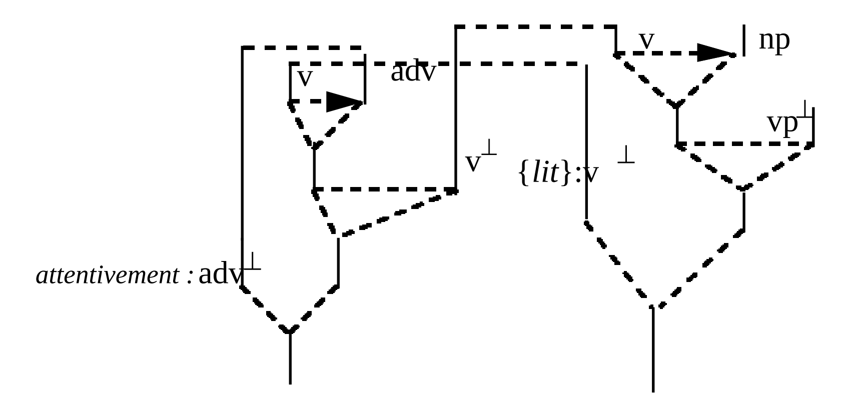


Branchement par coupure :



Élimination de la coupure :

(7c)



5. Conclusion

L'approche logique présentée ici n'est pas en soi une "théorie linguistique" : il s'agit de fournir un cadre formel qui peut permettre une expression adéquate de diverses théories linguistiques contemporaines, dont certains aspects de la théorie chomskyenne incluse dans le "Programme Minimaliste"[3]. Elle rejoint des tentatives (Joshi & Kulick, 1995, Abrusci, Fouqueré, Vauzeilles, 1996) pour jeter un pont entre des modèles à base d'arbres adjoints (TAGs) et le paradigme "dérivationnel" (ou "preuve comme calcul").

On peut ainsi rapprocher ce travail de celui de Abrusci, Fouqueré et Vauzeilles (désormais AFV) (Abrusci et al. 1996) qui utilise la logique linéaire intuitionniste non commutative (N-LLI) (qui s'obtient à partir du système LLI (§3-1) en supprimant la règle d'échange [4]). Abrusci et al. prouvent que tout arbre d'une grammaire TAG *G* peut être représentée par un séquent prouvable de N-ILL et que, si on considère, dans N-ILL, la fermeture de ces séquents sous *règle de coupure* et une règle dérivée du système que les auteurs nomment *règle d'adjonction*, alors cette fermeture correspond à la traduction dans N-ILL de l'ensemble des arbres obtenus par substitution et adjonction des arbres initiaux de *G*. Leur règle d'adjonction est la suivante :

$$\frac{\Gamma_1, a, \Gamma_2 \mid\!- a \qquad\qquad \Delta, a \text{ o-- } a, \Lambda \mid\!- b}{\Delta, \Gamma_1, \Lambda_1, \Gamma_2, \Lambda_2 \mid\!- b}$$

où (Λ_1, Λ_2) est la paire issue de manière unique de Λ telle que $\Lambda_1 \mid\!- a$ et $\Delta, a, \Lambda_2 \mid\!- b$

Elle se dérive des règles de N-ILL en appliquant deux fois la règle de coupure restreinte aux atomes. Dans un séquent associé à un arbre TAG, la formule A o-- A apparaît toujours pour indiquer que A est un noeud où l'adjonction (au sens TAG) peut avoir lieu. Ainsi par exemple, les arbres (6a) et (6b) se traduisent respectivement par :

(8a) VPo--V⊗NP, Vo--V, Vo-- *lit*, *lit*, NP |- VP

(8b) Vo--V⊗ADV, ADVo--*attentivement*, V, *attentivement* |- V

La règle dite d'adjonction permet de faire la déduction :

---

[3] Les formules que nous associons aux items lexicaux dans ce résumé sont assez "naïves" linguistiquement parlant, ceci à des fins d'exposition. Rien n'empêche évidemment d'avoir des formules beaucoup plus proches des hypothèses faites dans Chomsky, 1996. En particulier, il nous paraît clair qu'une théorie linguistique convenable doit éviter le genre de redondance que l'on a en introduisant des contraintes d'ordre dans chaque entrée lexicale. Les conditions d'ordre doivent découler de principes généraux. Pour cela, il suffira de procéder à la manière de Stabler, 1996, c'est-à-dire d'utiliser des catégories, pour nous: des modules, qui servent uniquement à l'assemblage des modules lexicalement pleins. De tels modules correspondent à AGR, T ou C, c'est-à-dire aux catégories fonctionnelles. Nous pourrons alors faire en sorte que seules ces modules puissent introduire de l'ordre.

[4] ce qui a pour effet de donner deux implications linéaires au lieu d'une, ici notées --o et o-- (cf. §3-4)

Vo--V⊗ADV, ADV o--*attentivement*, V, *attentivement* |- V   VPo--V⊗NP, Vo--V, Vo-- *lit*, *lit*, NP |- VP

VPo--V⊗NP, Vo--V⊗ADV, ADV o--*attentivement*, Vo-- *lit*, *lit*, *attentivement* , NP |- VP

qui correspond exactement à l'adjonction au sens de TAG (6c).

Comme on peut le constater, notre traduction des TAGs diffère de celle de AFV en ce que, dans cette dernière, l'adjonction se traduit en deux applications de la règle de coupure restreinte aux atomes ("condensées" dans la seule règle dite d'adjonction) alors que dans la notre elle se traduit en une seule coupure mais entre des formules quelconques. D'autre part, dans ADV, les arbres sont traduits en *séquents prouvables* alors que dans notre approche, ils le sont en *preuves partielles*.

A cause de ce dernier point, notre système se rapprocherait plutôt de celui de Joshi et Kulick (Joshi & Kulick, 1995, 1997). Dans ce dernier en effet, les auteurs obtiennent l'équivalent des TAGs en combinant des preuves partielles valides en grammaire catégorielle. Ils appellent *stretching* le fait de faire apparaître dans une preuve partielle la possibilité d'insérer une autre preuve : cela correspond exactement à notre introduction de conclusions supplémentaires afin de les relier par coupure, l'insertion même de la preuve correspondant alors au branchement par coupure et à son élimination. Ajoutons que notre système étant basé sur une logique des ordres partiels, il permet d'exprimer simplement la variabilité de l'ordre des mots dans certaines langues (le turc par exemple) ou dans certaines constructions syntaxiques (topicalisation).

Travailler avec des preuves partielles présente plusieurs intérêts par rapport à l'approche déductive classique. D'abord, sur le plan de l'efficacité, cela évite de refaire pour chaque analyse, certaines suites de pas déductifs qui sont ainsi "compactées" une bonne fois, cela évite donc certains problèmes d'ambiguïté superflue (*spurious ambiguity*) propres aux grammaires catégorielles. Ensuite, sur le plan théorique, comme dans les TAGs, il devient aisé d'exprimer les phénomènes de liage en introduisant des liens de coréférence dans les preuves partielles initiales et, plus généralement, d'exprimer des relations à distance arbitraire sous formes de phénomènes locaux (obtenant ce que A. Joshi définit comme un *domaine de localité étendu* (EDL), qui va de paire avec une mise en facteur de la récursion dans le domaine des dépendances)[5].

Il serait possible de montrer également que des formalismes dérivés des TAGs comme les "D-tree grammars" de Rambow et al. (Rambow et al. 1995) peuvent s'exprimer dans notre formalisme. D'une manière générale, on gagne à cet expression le fait de pouvoir caractériser les structures correctes par un critère topologique extrêmement simple (absence de circuit alterné). De plus, les réseaux ont une interprétation computationnelle intéressante en termes de parallélisme, qui permet d'établir un lien entre cette approche et une certaine variante du connexionnisme (Lecomte, 1994, Laks, 1996).

6. Bibliographie

Abrusci, M., Fouqueré, C. & Vauzeilles, J. (1996) : "Tree Adjoining Grammars in Noncommutative Linear Logic", in *Actes du Colloque LACL 96*, Nancy, septembre 1996

Berwick & Epstein (1996) : "On the Convergence Of ‘Minimalist’ Syntax and Categorial Grammar"

Chomsky, N. (1981) : *Lectures on Government and Binding*, Foris Pub.

---

[5] D'une manière générale, et ceci pour répondre à une remarque de l'un des lecteurs anonymes de cet article, c'est un avantage des grammaires lexicalistes de pouvoir exprimer certains phénomènes grammaticaux d'une manière beaucoup plus simple et adéquate que ne le font des grammaires faisant usage de règles et de principes plus généraux. C'est notablement le cas pour les phénomènes de liage et en particulier pour les anaphores (réflexifs et réciproques). De nombreux travaux (Reinhart & Reuland, 1993, Pollard et Sag, 1992) conduisent à considérer que le liage des réflexifs et des réciproques, étant un phénomène strictement local, a une explication naturelle dans le fait qu'il est spécifié dans le lexique. Baschung, 1991 fait la même observation à propos du contrôle des infinitives. Il n'en reste pas moins que, dans d'autres cas, un maximum d'information peut et doit être factorisé afin de ne pas réapparaître de façon excessivement redondante dans les entrées lexicales. C'est le sens de nos travaux futurs (voir aussi note 3).

Chomsky, N. (1996) : *The Minimalist Program,* The MITPress

Dalrymple, M. et al. (1995) : LFG Semantics, *in Proceedings of the ESSLLI Workshop on Formal Grammar*, Barcelone, aout 1995

Gazdar, G. al. (1985) : *Generalized Phrase Structure Grammar,* Blackwell

de Groote, P. & Retoré, C. (1996) : "On the Semantic Reading of Proof-Nets" in *Proceedings of the ESLLI Workshop on Formal Grammar,* Prague, août 1996

Girard, J.Y. (1987) *Linear Logic*, TCS, n°50

Girard, J.Y., Lafont, Y. & Régnier, L (1995) : *Advances in Linear Logic*, LMS

Johnson, M. (1997) : "LFG as a Resource Sensitive Grammar", rapport de recherches RXRC, Grenoble

Joshi, A. & Kulick, S. (1995) : "Partial Proof Trees as Building Blocks in a Categorial Grammar", in Morrill, G. et Oehrle, R (eds) *Formal Grammar, Proceedings of the ESSLLI Workshop* Barcelone, août 1995

Joshi, A. & Kulick, S. (1997) : "Partial Proof Trees, Resource Sensitive Logics and Syntactic Constraints", à paraître dans *Actes du Colloque LACL 96*, Nancy, septembre 1996

Laks, B. (1996) : *Langage et Cognition,* une approche connexionniste du langage, Hermes, Paris

Lecomte, A. (1992) : "Proof Nets and Dependencies", Actes de COLING 92, Nantes

Lecomte, A. (1993) : "Towards Efficient Parsing via Proof Nets", *Actes de EACL,* Utrecht

Lecomte, A. (1994) : *Modèles logiques en théorie linguistique,* mémoire d'habilitation, Grenoble

Lecomte, A. (1996) : "Grammaire et Théorie de la Preuve : une introduction", TAL, vol 37, n°2, numéro spécial sur *Grammaire et Théorie de la Preuve*

Lecomte, A. & Retoré, C. (1995) : "Pomset logic as an alternative categorial grammar" in Morrill, G. et Oehrle, R (eds) *Formal Grammar, Proceedings of the ESSLLI Workshop on Formal Grammar,* Barcelone

Lecomte, A. et Retoré, C. (1997) : "Words as Modules : a Lexicalized Grammar in the framework of Linear Logic Proof-nets", à paraître dans *Proceedings of ICML'96*, John Benjamins

Rambow, O., Vijay-Shanker, K., Weir, D. (1995) : "D-tree grammars", *33rd Meeting of the ACL*

Retoré, C. (1993) : *Calcul des Séquents Ordonnés*, thèse Paris VII

Retoré, C. (1996a) : "Perfect matchings and series-parallel graphs: multiplicative proof-nets as R&B-graphs", in Girard J.Y. et al. (eds) *Linear'96,* Electronic Notes in Computer Science, vol 3

Retoré, C. (1996b) : "Le calcul de Lambek et les réseaux de preuve", TAL, vol 37, n°2, numéro spécial sur *Grammaire et Théorie de la Preuve*

Retoré, C. (1997) : "Pomset logic : a non-commutative extension of classical linear logic", in J.R. Hindley et P. de Groote (eds) *TLCA'97* (LNCS vol 1210, Springer-Verlag)

Roorda, D. (1991) : *Resource Logics : proof-theoretical investigations*. PhD thesis, Université d'Amsterdam

Stabler, E. (1996) : *Acquiring and Parsing Languages with Movement*

Stabler, E. (1997) : "Derivational Minimalism", à paraître dans *Actes du Colloque LACL 96,* Nancy, septembre 1996

Steedman, M. (1996) : *Surface Structure and Interpretation,* MIT Press